Fuzzy Subspace Hidden Markov Models for Pattern Recognition

Dat Tran, Wanli Ma, and Dharmendra Sharma
Faculty of Information Sciences and Engineering
University of Canberra
ACT 2601, Australia
{dat.tran, wanli.ma, dharmendra.sharma}@canberra.edu.au

***Abstract*—This paper presents a novel fuzzy subspace-based approach to hidden Markov model. Features extracted from patterns are considered as feature vectors in a multi-dimensional feature space. Current hidden Markov modeling techniques treat features equally, however this assumption may not be true. We propose to consider subspaces in the feature space and assign a weight to each feature to determine the contribution of that feature in different subspaces to modeling and recognizing patterns. Weights can be computed if a learning estimation method such as maximum likelihood is given. Experimental results in network intrusion detection based on the proposed approach show promising results.**

***Keywords*: subspace, hidden Markov model, pattern recognition, network intrusion detection.**

I. INTRODUCTION

In statistical pattern recognition, hidden Markov model (HMM) is the most important technique for modeling patterns that include temporal information such as speech and handwriting. If the temporal information is not taken into account, Gaussian mixture model (GMM) is used. The GMM technique uses a mixture of Gaussian densities to model the distribution of feature vectors extracted from training data. The GMM technique is also regarded as the 1-state continuous HMM technique. When little training data are available, vector quantization (VQ) technique is also effective [32]. The VQ technique is regarded as a special case of the GMM technique if covariance matrices of Gaussian have the same constant values. In fuzzy set theory-based pattern recognition, fuzzy clustering techniques such as fuzzy c-means and fuzzy entropy are used to design re-estimation algorithms for fuzzy HMM, fuzzy GMM, and fuzzy VQ [33].

The first stage in modeling and recognizing patterns is data feature selection. A number of features that best characterizes the considering pattern is extracted and the selection of features is dependent on the pattern to be recognized and has direct impact on the recognition results. All of the above-mentioned pattern recognition methods cannot select features automatically and they also treat all features equally. We propose that the contribution of a feature to pattern recognition should be measured by a weight that is assigned to the feature in the modeling process. This method is called fuzzy subspace pattern recognition. There have been some algorithms proposed to calculate weights for fuzzy subspace clustering [9, 11]. However a generic framework that can apply to HMM, GMM, and VQ modeling techniques does not exist.

In this paper, we propose a novel fuzzy subspace-based approach that can apply to all of the above-mentioned techniques. We consider the pattern recognition problem in maximum likelihood criterion. A generic objective function based on maximum likelihood and fuzzy c-means estimation is designed for the fuzzy subspace HMM and maximizing this function will result in an algorithm for calculating weights as well as HMM parameters. Algorithms for the fuzzy subspace GMM and VQ techniques will also be determined from the algorithm for the fuzzy subspace HMM.

The proposed fuzzy subspace pattern recognition methods will be evaluated in network intrusion detection. Some preliminary experiments have been done and experimental results showed that the proposed approach can improve the recognition rates.

II. FUZZY SUBSPACE HIDDEN MARKOV MODEL

The underlying assumption of the HMM is that the considering pattern can be well characterized as a parametric random process, and that the parameters of the stochastic process can be estimated in a precise, well-defined manner. The HMM technique provides a reliable way of recognizing speech for a wide range of applications [8, 12, 23].

There are two assumptions in the first-order HMM. The first one is the Markov assumption, i.e. a new state is entered at each time t based on the transition probability, which only depends on the previous state. It is used to characterize the sequence of the time frames of a pattern. The second is the output-independence assumption, i.e. the output probability depends only on the state at that time regardless of when and how the state is entered [10]. A process satisfying the Markov assumption is called a Markov model [15]. An observable Markov model is a process where the output is a set of states at each instant of time and each state corresponds to an observable event. The hidden Markov model is a doubly stochastic process with an underlying Markov process which is not directly observable (hidden) but which can be observed

978-1-4244-4568-4/09/$25.00 ©2009 IEEE

through another set of stochastic processes that produce observable events in each of the states [24].

Let $S = \{s_1, s_2, ..., s_T\}$ and $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_T\}$ be a sequence of states and a sequence of continuous feature vectors, respectively. The compact notation $\Lambda = \{\pi, A, B\}$ indicates the complete parameter set of the HMM where

- $\pi = \{\pi_i\}$, $\pi_i = P(s_1 = i \mid \Lambda)$: the initial state distribution
- $A = \{a_{ij}\}$, $a_{ij} = P(s_t = j \mid s_{t-1} = i, \Lambda)$: the state transition probability distribution, and
- $B = \{b_j(\mathbf{x}_t)\}$, $b_j(\mathbf{x}_t) = P(\mathbf{x}_t \mid s_t = j, \Lambda)$: the output probability distribution of feature vector $\mathbf{x}_t$ in state j.

The following constraints are applied:

$$\sum_{i=1}^{N} \pi_i = 1, \quad \sum_{j=1}^{N} a_{ij} = 1, \text{ and } \int b(\mathbf{x}_t) d\mathbf{x}_t = 1 \qquad (1)$$

The HMM parameters are estimated such that in some sense, they best match the distribution of the feature vectors in $\mathbf{X}$. The most widely used training method is the maximum likelihood (ML) estimation. For a sequence of feature vectors $\mathbf{X}$, the likelihood of the HMM is

$$P(\mathbf{X} \mid \Lambda) = \prod_{t=1}^{T} P(\mathbf{x}_t \mid \Lambda) \qquad (2)$$

The aim of ML estimation is to find a new parameter model $\overline{\Lambda}$ such that $P(\mathbf{X} \mid \overline{\Lambda}) \geq P(\mathbf{X} \mid \Lambda)$. Since the expression in (2) is a nonlinear function of parameters in Λ, its direct maximisation is not possible. However, parameters can be obtained iteratively using the expectation-maximization (EM) algorithm [6]. An auxiliary function Q is used

$$Q(\Lambda, \overline{\Lambda}) = \sum_{t=1}^{T-1} \sum_{i=1}^{N} \sum_{j=1}^{N} P(s_t = i, s_{t+1} = j \mid \mathbf{X}, \Lambda) \log[\overline{a}_{ij} \overline{b}_{ij}(\mathbf{x}_{t+1})] \qquad (3)$$

where $\overline{\pi}_{s_1 = j}$ is denoted by $\overline{a}_{s_0 = i\, s_1 = j}$ for simplicity. The most general representation of the output probability distribution is a mixture of Gaussians

$$b_j(\mathbf{x}_t) = P(\mathbf{x}_t \mid s_t = j, \Lambda) = \sum_{k=1}^{K} P(k \mid s_t = j, \Lambda) P(\mathbf{x}_t \mid k, s_t = j, \Lambda) \qquad (6)$$

This can be rewritten as

$$b_j(\mathbf{x}_t) = \sum_{k=1}^{K} c_{jk} N(\mathbf{x}_t, \boldsymbol{\mu}_{jk}, \boldsymbol{\Sigma}_{jk}) \qquad (7)$$

where $c_{jk} = P(k \mid s_t = j, \Lambda)$, $j = 1, ..., N$, $k = 1, ..., K$ are mixture coefficients, and $N(\mathbf{x}_t, \boldsymbol{\mu}_{jk}, \boldsymbol{\Sigma}_{jk})$ is a Gaussian with mean vector $\boldsymbol{\mu}_{jk}$ and covariance matrix $\boldsymbol{\Sigma}_{jk}$ for the k-th mixture component in state j. The following constraints need to be satisfied

$$c_{jk} > 0 \qquad \text{and} \qquad \sum_{k=1}^{K} c_{jk} = 1 \qquad (8)$$

In order to differentiate the contribution of features, we propose to assign a weight w_{jkm}^{α} to the m-th feature as follows

$$\log N(\mathbf{x}_t, \overline{\boldsymbol{\mu}}_{jk}, \overline{\boldsymbol{\Sigma}}_{jk}) = \sum_{m=1}^{M} w_{jkm}^{\alpha} \log P(x_{tm} \mid k, s_t = j, \overline{\Lambda}) \qquad (9)$$

where

$$P(x_{tm} \mid k, s_t = j, \Lambda) = \frac{1}{\sqrt{2\pi\sigma_{jkm}^2}} e^{-\frac{(x_{tm} - \mu_{jkm})^2}{2\sigma_{jkm}^2}} \qquad (10)$$

σ_{jkm} is the m-th variance component in Gaussian k and state j, w_{jkm}^{α}, $m = 1, 2, \ldots, M$ are components of an M-dimensional weight vector $\mathbf{w}_{jm}^{\alpha}$, and α is a fuzzy parameter weight for w_{jkm}^{α}. The weight values satisfy the following conditions:

$$0 \leq w_{jkm} \leq 1 \quad \forall m, \qquad \sum_{m=1}^{M} w_{jkm} = 1 \qquad (11)$$

It can be seen that if all of the weight values are equal, the proposed expression of Gaussian distribution $N(\mathbf{x}_t, \overline{\boldsymbol{\mu}}_{jk}, \overline{\boldsymbol{\Sigma}}_{jk})$ in (9) becomes the normal expression for Gaussian distribution as seen in statistics and probability theory [10].

Maximizing the likelihood function in (2) can be obtained by maximizing the objective function in (3) over $\overline{\Lambda}$ and the weight vector $\mathbf{w}_{jm}^{\alpha}$. The basic idea of this fuzzy subspace-based approach is the function $Q_j(\Lambda, \overline{\Lambda})$ is maximized over the variable w_{jkm} on the assumption that the weight vector $\mathbf{w}_{jm}$ identifies a good contribution of the features. Using the well-known Lagrange multiplier method, maximizing the function $Q_j(\Lambda, \overline{\Lambda})$ in (3) using (8) and (11) gives

$$w_{jkm} = \frac{1}{\sum_{n=1}^{M} (D_{jkm} / D_{jkn})^{1/(\alpha-1)}} \qquad (12)$$

where $\alpha \neq 1$ and

$$D_{jkm} = -\sum_{t=1}^{T} P(k \mid \mathbf{x}_t, s_t = j, \Lambda) \log P(x_{tm} \mid k, s_t = j, \overline{\Lambda}) \quad (13)$$

The mixture coefficients, mean vectors and covariance matrices are calculated by maximizing the function in (3) over $\overline{\Lambda}$ using (1) and (8). We obtain:

$$\overline{c}_{jk} = \frac{1}{T} \sum_{t=1}^{T} P(k \mid \mathbf{x}_t, s_t = j, \Lambda) \quad (14)$$

$$\overline{\boldsymbol{\mu}}_{jk} = \frac{\sum_{t=1}^{T} P(k \mid \mathbf{x}_t, s_t = j, \Lambda) \mathbf{x}_t}{\sum_{t=1}^{T} P(k \mid \mathbf{x}_t, s_t = j, \Lambda)} \quad (15)$$

$$\overline{\boldsymbol{\Sigma}}_{jk} = \frac{\sum_{t=1}^{T} P(k \mid \mathbf{x}_t, s_t = j, \Lambda)(\mathbf{x}_t - \boldsymbol{\mu}_{jk})(\mathbf{x}_t - \boldsymbol{\mu}_{jk})'}{\sum_{t=1}^{T} P(k \mid \mathbf{x}_t, s_t = j, \Lambda)} \quad (16)$$

where the prime denotes vector transposition, and

$$P(k \mid \mathbf{x}_t, s_t = j, \Lambda) = \frac{c_{jk} N(\mathbf{x}_t, \boldsymbol{\mu}_{jk}, \boldsymbol{\Sigma}_{jk})}{\sum_{n=1}^{K} c_{jn} N(\mathbf{x}_t, \boldsymbol{\mu}_{jn}, \boldsymbol{\Sigma}_{jn})} \quad (17)$$

The initial state distribution and state transition distribution are also determined:

$$\overline{\pi}_i = \gamma_1(i) \qquad \overline{a}_{ij} = \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t(i)} \quad (18)$$

where

$$\gamma_t(i) = \sum_{j=1}^{N} \xi_t(i,j), \quad \xi_t(i,j) = P(s_t = i, s_{t+1} = j \mid \mathbf{X}, \Lambda) \quad (19)$$

The advantage of this approach is that when the weighting values w_{jkm}^{α} have the same value, the fuzzy subspace-based HMM becomes the standard HMM in the maximum likelihood criterion. Therefore, the proposed approach can be considered as a generic framework and can extend to other models that relate to the HMM such as GMM and VQ, and other criteria such as minimum classification error (MCE) and maximum a posteriori (MAP).

III. Fuzzy Subspace Gaussian Mixture Model

Fuzzy subspace GMM can be obtained by setting the number of states in fuzzy subspace continuous HMM to one. The GMM parameters consist of the mixture weight c_{jk}, mean vector $\boldsymbol{\mu}_{jk}$, covariance matrix $\boldsymbol{\Sigma}_{jk}$, and subspace weight $\mathbf{w}_{jm}$. The estimation equations in (12), (14), (15), (16), and (17) are used to calculate the GMM parameters.

IV. Fuzzy Subspace Vector Quantization

The VQ modeling is an efficient data reduction method, which is used to convert a feature vector set into a small set of distinct vectors using a clustering technique. Advantages of this reduction are reduced storage and computation. The distinct vectors are called code vectors and the set of code vectors that best represents the training set is called the codebook. Since there is only a finite number of code vectors, the process of choosing the best representation of a given feature vector is equivalent to quantizing the vector and leads to a certain level of quantization error. This error decreases as the size of the codebook increases, however the storage required for a large codebook is non-trivial. The VQ codebook can be used as a model in pattern recognition. The key point of VQ modeling is to derive an optimal codebook which is commonly achieved by using a clustering technique.

In VQ modeling, the model Λ is a set of cluster centers $\Lambda = \{\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, ..., \boldsymbol{\mu}_K\}$ where $\boldsymbol{\mu}_k = (\mu_{k1}, \mu_{k2}, ..., \mu_{kM})$, k = 1, 2, …, K are code vectors (also mean vectors). Each code vector $\boldsymbol{\mu}_k$ is assigned to an encoding region R_k in the partition $\Omega = \{R_1, R_2, ..., R_K\}$. Then the source vector $\mathbf{x}_t$ can be represented by the encoding region R_k and expressed by

$$V(\mathbf{x}_t) = \boldsymbol{\mu}_k \qquad \text{if} \qquad \mathbf{x}_t \in R_k \quad (20)$$

Let $U = [u_{kt}]$ be a matrix whose elements are memberships of x_t in the n-th cluster, k = 1, 2, …, K, t = 1, 2, …, T. A K-partition space for **X** is the set of matrices U such that

$$u_{kt} \in \{0,1\} \ \forall k,t, \quad \sum_{k=1}^{K} u_{kt} = 1 \ \forall t, \quad 0 < \sum_{t=1}^{T} u_{kt} < T \ \forall k \quad (21)$$

where $u_{kt} = u_k(\mathbf{x}_t)$ is 1 or 0, according to whether $\mathbf{x}_t$ is or is not in the kth cluster, $\sum_{k=1}^{K} u_{kt} = 1 \ \forall t$ means each $\mathbf{x}_t$ is in exactly one of the K clusters, and $0 < \sum_{t=1}^{T} u_{kt} < T \ \forall k$ means that no cluster is empty and no cluster is all of **X** because of 1 < K < T.

The fuzzy subspace VQ technique is based on minimization of the $J(U, W, \Lambda)$ function obtained from the $Q(\Lambda, \overline{\Lambda})$ function in (3) by removing the expressions that contains state parameters in HMM and Gaussian parameters in GMM. The $J(U, W, \Lambda)$ function is also considered as the sum-of-squared-errors function (the index j for state is omitted) as follows

$$J(U,W,\Lambda) = \sum_{k=1}^{K}\sum_{t=1}^{T} u_{kt} \sum_{m=1}^{M} w_{km}^{\alpha} d_{ktm} \quad (22)$$

where $\overline{\Lambda}$ is included in d_{ktm}, which is the Euclidean norm of $(\mathbf{x}_t - \boldsymbol{\mu}_k)$. Similarly, the well-known Lagrange multiplier method is used to obtain the following equations for fuzzy subspace VQ

$$\boldsymbol{\mu}_k = \sum_{t=1}^{T} u_{kt}\mathbf{x}_t \Big/ \sum_{t=1}^{T} u_{kt}, \quad 1 \le k \le K \quad (23)$$

$$u_{kt} = \begin{cases} 1: & d_{kt} < d_{jt}, \quad j = 1,...,K, j \ne k \\ 0: & otherwise \end{cases} \quad (24)$$

$$w_{km} = \frac{1}{\sum_{n=1}^{M} (D_{km}/D_{kn})^{1/(\alpha-1)}}, \qquad D_{km} = \sum_{t=1}^{T} u_{kt} d_{ktm} \quad (25)$$

where

$$d_{ktm} = (c_{km} - x_{tm})^2, \qquad d_{kt} = \sum_{m=1}^{M} w_{km}^{\alpha} d_{ktm}^2 \quad (26)$$

V. Algorithms for Modeling and Recognizing Patterns

A. Modeling Algorithm

The modeling algorithm for the fuzzy subspace HMM technique is summarized as follows:

1. Give a training data set $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_T\}$, where $\mathbf{x}_t = (x_{t1}, x_{t2}, ..., x_{tM})$, t = 1, 2,…, T.
2. Initialize parameters at random satisfying (1), (8) and (11)
3. Give $\alpha \ne 1$ and $\varepsilon > 0$ (small real number)
4. Set i = 0 and $Q^{(i)}(\Lambda, \overline{\Lambda})$ to a small real number. Iteration:
   a. Compute weight values using (12) and (13)
   b. Compute Gaussian parameters using (14), (15), (16), and (17)
   c. Compute state parameters using (18) and (19)
   d. Compute $Q^{(i+1)}(\Lambda, \overline{\Lambda})$ using (3), (6), (7), (9), and (10)
   e. If

$$\frac{Q^{(i+1)}(\Lambda, \overline{\Lambda}) - Q^{(i)}(\Lambda, \overline{\Lambda})}{Q^{(i+1)}(\Lambda, \overline{\Lambda})} < \varepsilon \quad (27)$$

   set $Q^{(i)}(\Lambda, \overline{\Lambda}) = Q^{(i+1)}(\Lambda, \overline{\Lambda})$, $i = i + 1$ and go to step (a).

B. Recognition Algorithm

Assuming $\{\Lambda^{(1)}, \Lambda^{(2)}, ..., \Lambda^{(p)}\}$ are p pattern models that are trained using the modeling algorithm. Given an unknown feature vector $\mathbf{x}$, the task is to classify $\mathbf{x}$ into one of the p models. The following algorithm is proposed

1. Given an unknown feature vector $\mathbf{x}$ and the set of models $\{\Lambda^{(1)}, \Lambda^{(2)}, ..., \Lambda^{(p)}\}$
2. Calculate the probabilities $P(\mathbf{x} \mid \Lambda^{(i)})$, i = 1, …, p.
3. The recognized model i^* is the model whose probability is maximum

$$i^* = \arg\max_{i \in \{1,2,...,p\}} P(\mathbf{x} \mid \Lambda^{(i)}) \quad (28)$$

VI. Experimental Results

We used the KDD CUP 1999 dataset [13] to evaluate the proposed approach. This dataset was based on MIT Lincoln Lab intrusion detection dataset, also known as DARPA dataset [5]. The data was produced for "The Third International Knowledge Discovery and Data Mining Tools Competition", which was held in conjunction with the Fifth International Conference on Knowledge Discovery and Data Mining. The raw network traffic records have already been converted into vector format. Each feature vector consists of 41 features. The meanings of these features can be found in [30].

The attacks listed in feature vectors of KDD CUP 1999 dataset come from MIT Lincoln intrusion detection dataset web site (KDD CUP 1999). The labels are mostly the same except a few discrepancies. The MIT Lincoln lab web site lists 2 types of buffer overflow attack: *eject* and *ffb*. The former explores the buffer overflow problem of *eject* program of Solaris, and the later explores the buffer overflow problem of *ffb* config program. Guessing user logon names and passwords through remote logon via telnet session is labeled as *guess_passwd* in the KDD CUP 1999 dataset, but listed as *dict* on the MIT Lincoln lab web site. Finally, we cannot find the counterparts of *syslog* and *warez* in the KDD CUP 1999 dataset. In addition to the attack labels, the KDD CUP 1999 dataset has also the label *normal*, which means that the traffic is normal and free from any attack.

The proposed method for network intrusion detection was evaluated using the KDD CUP 1999 data set for training and the *Corrected* data set for testing. For training, the number of feature vectors for training the *normal* model was set to 5000. For testing, there were not sufficient data for all attack types, so we selected the *normal* network pattern and the 5 attacks which were *ipsweep*, *neptune*, *portsweep*, *satan*, and *smurf*. The testing data set contains 60593 feature vectors for the *normal* network pattern, and 306, 58001, 354, 1633 and 164091 feature vectors for the five attacks, respectively.

We also conducted a set of experiments for the network data using the normalization technique as follows

$$x'_{tm} = \frac{x_{tm} - \mu_m}{\sigma_m}, \quad \sigma_m = \frac{1}{T}\sum_{t=1}^{T} | x_{tm} - \mu_m | \qquad (29)$$

where x_{tm} is the m-th feature of the t-th feature vector, μ_m the mean value of all T feature vectors for feature m, and σ_m the mean absolute deviation.

Anomaly detection rates versus false alarm rates are presented in Tables 1, 2, 3, and 4, where the number of Gaussians is set to 4, 8, 16, and 32, respectively. The value of α was set to 4. All network data were normalized. We chose 5 false alarm rates (in %) which were 0.0, 0.1, 1.0, 10.0, and 100.0 to compare the corresponding anomaly detection rates for the standard GMM modeling and the proposed fuzzy subspace GMM modeling method. The ideal value for false alarm rate is 0.0, and from the 4 tables, we can see that the fuzzy subspace GMM performed outperformed the standard GMM modeling even with the smallest Gaussians.

All the considered methods could not achieve the highest anomaly detection rate of 100% even though we changed the threshold value to accept all attack patterns (i.e., the false alarm rate is 100%). With 32 Gaussians, the fuzzy subspace GMM modeling achieved very good results even with the lowest false alarm rate. The training data set contained 5000 feature vectors. If all training data for the *normal* pattern were used to train the model, the result would be better.

TABLE I. ANOMALY DETECTION RESULTS, #GAUSSIANS = 4

Modeling	False Alarm Rate (in %)				
	0.0	0.1	1.0	10.0	100.0
GMM	45.8	46.2	46.9	48.7	77.6
Fuzzy Subspace GMM	98.0	98.3	98.4	98.6	98.8

TABLE II. ANOMALY DETECTION RESULTS, #GAUSSIANS = 8

Modeling	False Alarm Rate (in %)				
	0.0	0.1	1.0	10.0	100.0
GMM	46.2	50.1	53.2	60.1	78.1
Fuzzy Subspace GMM	98.1	98.3	98.5	98.6	98.9

TABLE III. ANOMALY DETECTION RESULTS, #GAUSSIANS = 16

Modeling	False Alarm Rate (in %)				
	0.0	0.1	1.0	10.0	100.0
GMM	65.9	80.2	82.5	84.3	91.8
Fuzzy Subspace GMM	98.1	98.3	98.6	98.6	99.1

TABLE IV. ANOMALY DETECTION RESULTS, #GAUSSIANS = 32

Modeling	False Alarm Rate (in %)				
	0.0	0.1	1.0	10.0	100.0
GMM	83.1	84.1	86.1	87.0	95.3
Fuzzy Subspace GMM	98.6	99.0	99.1	99.2	99.5

VII. CONCLUSION

We have proposed a generic framework for fuzzy subspace-based methods in pattern recognition. The framework has been presented for fuzzy subspace HMM using maximum likelihood estimation. We have also presented how the fuzzy subspace GMM and fuzzy subspace VQ can be obtained from the fuzzy subspace HMM. We have also applied the proposed methods to anomaly network detection and evaluated the methods with the KDD CUP 1999 dataset. The results in anomaly network detection showed that the fuzzy subspace HMM, GMM, and VQ can be used in pattern recognition. The selection of useful features is a very important task for any classifier and is worth investigating.

REFERENCES

[1] R. Anderson and A. Khattak, "The use of Information Retrieval Techniques for Intrusion Detection", in Proceedings of the first International Workshop on Recent Advances in Intrusion Detection (RAID'98), Louvain-la-Neuve, Belgium, 1998.

[2] J.S. Balasubramaniyan, J.O. Garcia-Fernandez, et al., "An Architecture for Intrusion Detection using Autonomous Agents", in Proceedings of the 14th IEEE ACSAC, Scottsdale, AZ, USA, pp. 13-24, 1998.

[3] C. Caruso and D. Malerba, Clustering as an add-on for firewalls, Data Mining, WIT Press, 2004.

[4] P.K. Chan, M.V. Mahoney, and M.H. Arshad, A Machine Learning Approach to Anomaly Detection, Technical Report CS-2003-06, 2003.

[5] DARPA Intrusion Detection Evaluation Data Sets 1999, available at http://www.ll.mit.edu/IST/ideval/data/data_ index.html

[6] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum Likelihood from Incomplete Data via the EM algorithm", Journal of the Royal Statistical Society, Ser. B, 39: pp. 1-38, 1997.

[7] E. Eskin, "Anomaly Detection over Noisy Data Using Learned Probability Distributions", in the 17th International Conference on Machine Learning, Morgan Kaufmann, San Francisco, USA, pp. 255-262, 2000.

[8] S. Furui, "Recent advances in speaker recognition", *Patter Recognition Lett.*, vol. 18, pp. 859-872, 1997.

[9] J.Z. Huang, M.K. Ng, H. Rong, and Z. Li, "Automated Variable Weighting in k-means Type Clustering", *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 27, no. 5, pp. 657-668, 2005.

[10] X. Huang, A. Acero, F. Alleva, M. Huang, L. Jiang, and M. Mahajan, From SPHINX-II to WHISPER: Making speech recognition usable, chapter 20 in *Automatic Speech and Speaker Recognition, Advanced Topics*, edited by Chin-Hui Lee, Frank K. Soong, and Kuldip K. Paliwal, Kluwer Academic Publishers, USA, pp. 481-508, 1996.

[11] L. Jing, M. K. Ng., J. Z. Huang, "An entropy weighting k-means algorithm for subspace clustering of high-dimensional sparse data", *IEEE Transactions on Knowledge and Data Engineering*, vol. 19, no. 6, pp. 1026-1041, 2007.

[12] B.-H. Juang, The Past, Present, and Future of Speech Processing, *IEEE Signal Processing Magazine*, vol. 15, no. 3, pp. 24-48, 1998.

[13] KDD CUP 1999 Data Set, available at the following website http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

[14] Stanifor, Hoagland and McAlerney, "Practical Automated Detection of Stealthy PortScans", *Journal of Computer Security*, vol. 10, no. 1, pp. 105-136, 2002.

[15] V. G. Kulkarni, *Modeling and analysis of stochastic systems*, Chapman & Hall, UK, 1995.

[16] X. Li and N. Ye, "Mining Normal and Intrusive Activity Patterns for Computer Intrusion Detection", in Intelligence and Security Informatics: Second Symposium on Intelligence and Security Informatics, Tucson, USA, Springer-Verlag, vol. 3073, pp. 1611-3349, 2004.

[17] W. Lee and D. Xiang, Information theoretic measures for anomaly detection, in IEEE Synposium on Security and Privacy, pp. 130-143, 2001.

[18] M. V. Mahoney, and P.K. Chan, "PHAD: Packet Header Anomaly Detection for Identifying Hostile Network Traffic", Technical report, Florida Tech., CS-2001-4, 2001.

[19] M. Mahoney, "Network Traffic Anomaly Detection Based on Packet Bytes", Proc. ACM. Symposium on Applied Computing, pp. 346-350, 2003.

[20] D. Ourston, S. Matzner, et al., "Coordinated Internet attacks: responding to attack complexity", *Journal of Computer Security*, vol. 12, pp. 165-190, 2004.
[21] V. Paxson, "Bro: A system for detecting network intruders in real-time", in Proceedings of the 7th USENIX Security Symposium, Texas, USA, pp. 3-7, 1998.
[22] L. Portnoy, E. Eskin, and S. Stolfo, "Intrusion detection with unlabeled data using clustering", in Proceedings of ACM CSS Workshop on Data Mining Applied to Security (DMSA-2001), Philadelphia, USA, pp. 333-342, 2001.
[23] L. R. Rabiner, B. H. Juang and C. H. Lee, "An Overview of Automatic Speech Recognition", chapter 1 in Automatic Speech and Speaker Recognition, Advanced Topics, edited by Chin-Hui Lee, Frank K. Soong, and Kuldip K. Paliwal, Kluwer Academic Publishers, USA, pp. 1-30, 1996.
[24] L. R. Rabiner, and B. H. Juang, Fundamentals of speech recognition, Prentice Hall PTR, USA, 1993.
[25] J.S. Sherif, R. Ayers, and T. G. Dearmond, "Intrusion Detection: the art and the practice", Part 1. Information Management and Computer Security, vol. 11, no. 4, pp. 175-186, 2003.
[26] J.S. Sherif and R. Ayers, "Intrusion detection: methods and systems", Part II. Information Management and Computer Security, vol. 11, no. 5, pp. 222-229, 2003.
[27] S.J. Stolfo, W. Fan, W. Lee, A. Prodromidis, and P.K. Chan, "Cost-based Modeling and Evaluation for Data Mining With Application to Fraud and Intrusion Detection: Results from the JAM Project", in Proceedings of DARPA Information Survivability Conference and Exposition, 2000, pp. 1130-1144, 2000.
[28] C. Taylor and J. Alves-Foss, "An Empirical Analysis of NATE: Network Analysis of Anomalous Traffic Events", in 10th New Security Paradigms Workshop, Virginia Beach, Virginia, USA, pp. 18-26, 2002.
[29] C. Taylor and J. Alves-Foss, "NATE: Network Analysis of Anomalous Traffic Events, a low-cost approach", in Proceedings of New Security Paradigms Workshop, Cloudcroft, New Mexico, USA, pp. 89-96, 2001.
[30] Tran D., Ma W., Sharma D. and Nguyen T. (2007). Fuzzy Vector Quantization for Network Intrusion Detection, IEEE International Conference on Granular Computing, Silicon Valley, USA
[31] D. Tran, W. Ma, and D. Sharma, "Automated Feature Weighting for Network Anomaly Detection", IJCSNS International Journal of Computer Science and Network Security, Vol. 8 No. 2 pp. 173-178, 2008.
[32] D. Tran and M. Wagner, "Generalised Fuzzy Hidden Markov Models for Speech Recognition", Lecture Notes in Computer Science: Advances in Soft Computing - AFSS 2002, N.R. Pal, M. Sugeno (Eds.), pp. 345-351, Springer-Verlag, 2002.
[33] D. Tran and M. Wagner, "A General Approach to Hard, Fuzzy, and Probabilistic Models for Pattern Recognition", Advances in Intelligent Systems: Theory and Applications, M. Mohammadian (ed.), pp. 244-251, IOS Press, Netherlands, 2000.
[34] Y. Yasami, M. Farahmand, V. Zargari, "An ARP-based Anomaly Detection Algorithm Using Hidden Markov Model in Enterprise Networks", in Proc. Second International Conference on Systems and Networks Communications, pp. 69 – 75, 2007.
[35] H. Yang, F. Xie, and Y. Lu, "Clustering and Classification Based Anomaly Detection", Lecture Notes in Computer Science, vol. 4223, pp. 1611-3349, 2006.